Reliable Multicast Transport and Integrated Erasure-Based Forward Error Correction

Joseph P. Macker
Center for High Assurance Computer Systems
Naval Research Laboratory

macker@itd.nrl.navy.mil

Abstract

Multicast networking is an important emerging technology area for both commercial and military group-based data dissemination. In addition, a number of emerging applications can benefit from a reliable multicast transport service. A variety of approaches have been developed regarding the general application of Automatic Repeat Request (ARQ) techniques over Internet Protocol (IP) multicasting to achieve reliable delivery. In this paper, we investigate the application of erasure-based processing and parity-based recovery to a reliable multicast protocol framework. The integrated design approach described is shown to have improved efficiency and scalability features over reliable multicasting techniques based solely on ARQ. These bandwidth utilization improvements are expected to be substantial when applied across future multipoint communication infrastructures, especially over bandwidth-constrained and/or asymmetric networks.

Introduction

Reliable multicast protocol design must handle the problem of reliable ensuring delivery of data from a sender to *M* receivers despite packet loss that occurs within a network. The recovery of data loss within a communication system has classically been solved via two methods: Automatic Repeat Request (ARQ) and Forward Error Correction (FEC). ARQ involves receiver loss notification to the source and subsequent retransmission of lost data. In contrast, FEC approaches generally involve the repairing of lost data over the transmission channel by processing parity information sent along with the data transmission. Numerous reliable multicast networking protocols and frameworks based solely around ARQ techniques have been well developed and discussed in previous work. Until recently, little work has been done in the area of applying FEC methods to reliable multicasting transport mechanisms. The integration of FEC techniques, specifically erasure-based coding, can improve efficiency and scalability when combined appropriately with existing ARQ packet techniques. The resulting reduction in required consumption of network bandwidth while maintaining reliability improves the utility of future military and commercial group-based data dissemination applications.

As multicast receiver populations grow across an internetwork, the potential for lost packets generally increases. At present, many reliable multicast protocol approaches attempt to limit the amount of closed loop control traffic (i.e., improve scalability) by using selective repeat negative acknowledgment (NACK) retransmission methods. In addition, probabilistic delay backoff windows are often used to reduce implosion and redundant negative acknowledgments triggered by lost packets within a multicast receiver group [1,2]. To further improve reliable multicast performance there has been growing interest in the study of applying FEC and erasure processing techniques [3,4,5,6].

Many future wireless network channels are anticipated to have relatively high packet loss statistics (e.g., wireless networks) or they may additionally be asymmetric in nature (e.g., long delay, high data rate satellite downlinks in concert with low speed terrestrial wireless return channels). For these networks, it is further desirable to reduce the amount of required retransmission requests and the subsequent expected retransmission traffic as much as possible [8].

Here we discuss a novel approach to hybrid transport control that can be used as to improve reliable multicast data dissemination efficiency. In addition, this method can be applied to network unicast bulk transfer within multipoint delivery systems, but more significant performance value is achieved when applied to multicast network data transfer.

Background

The present underlying delivery mechanism for Internet Protocol (IP) multicast is presently the User Datagram Protocol (UDP) or raw IP packets, as largely defined in [9]. At present, these raw mechanisms provide a "best effort" delivery service. Best effort implies that IP packets are treated with essentially equal weight, and while IP makes an effort to deliver all packets to their destination, packets may be occasionally be delayed, lost, duplicated, or delivered out of order. In the past such delivery mechanisms have worked fine for supporting traffic insensitive to occasional lost or missing data (e.g.. voice, video). An increasing variety of distributed commercial and military applications are being developed in which a consistent and/or reliable data delivery of all or a subset of data packets is a critical performance factor [10]. The Internet Integrated Services Architecture (ISA) includes emerging technology components that can provide some Quality of Service (QoS) forwarding capabilities within an internetwork. This topic has been covered by other work and is not discussed here although its potential supportive role in improving reliable multicasting is recognized [11]. A variety of reliable multicast transport protocols and protocol frameworks have been proposed in recent years which can provide end to end reliability over best effort datagram service [1,2,13,14,15,17,18,19,21,22].

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **1997**	2. REPORT TYPE	3. DATES COVERED **00-00-1997 to 00-00-1997**

4. TITLE AND SUBTITLE **Reliable Multicast Transport and Integrated Erasure-based Forward Error Correction**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Research Laboratory,Center for High Assurance Computer Systems,4555 Overlook Avenue, SW,Washington,DC,20375**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**5**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Most reliable multicast file or bulk transfer applications are presently designed around the basic concept of selective repeat ARQ without the addition of any FEC coding. The source application divides the data to be transmitted into application data units (ADUs) or transport layer protocol data units (PDUs). Throughout this paper we refer to these units as packets. These packets are generally tagged with sequence numbers and additional information (e.g., source identification) allowing receivers to determine which ongoing transfer incoming data packets belong to and their relative ordering. This sequencing information can also be used independently by receivers to detect dropped packets or missing "gaps" in received data. In many reliable multicast protocols, receivers use this information to request retransmission for missing packets as required. Such receiver-based NACK reliability schemes are more scalable and efficient than source-oriented acknowledgment [10,16].

Several additional techniques have been adopted by a number of protocols to increase the scalability and efficiency of pure-ARQ approaches. In some cases, redundant retransmission requests and repairs within a multicast group can be dampened through appropriate usage of backoff windows and timers [1,2]. In other cases, hierarchies have been proposed to improve scalability through the selection or designation of intermediate or specialized multicast nodes within the group population [21,22].

Recent work on the Image Multicaster (IMM) built upon the Multicast Dissemination Protocol (MDP) framework is an example of a reliable multicast application that is receiver-based and uses NACK suppression within the receiver population [2]. It uses the concept of transmission blocks and repair cycles to achieve end to end reliability. While retransmission is an important feature for reliable networking, reducing the both the retransmission requests and subsequent packet retransmission is a critical performance enhancement. An integrated erasure-based ARQ/FEC approach will be shown to greatly reduce reliable multicast transport resource requirements under the following conditions.

- uncorrelated packet loss
- increasing receiver group size
- error-prone links (e.g., wireless, mobile, network congestion)

The idea of combining FEC and ARQ for multipoint communications has been studied by a number of authors at the time of writing [3,4,5,6,7]. Many of these papers have dealt with the idea of layering FEC below a reliable ARQ protocol. A study of this layered approach against an integrated approach has recently been presented under a variety of modeling and error conditions [6]. The study showed that integrated approaches have efficiency advantages over layered approaches. In this paper, we will consider such an example of an integrated design approach of combining erasure-based FEC coding and receiver-based reliable multicast ARQ. The next section of this paper will describe the method envisioned and will discuss related performance issues.

Method

We here present an integrated design approach to erasure-based FEC coding and reliable multicast transport. A number of points make these requirements unique in comparison to classic FEC design within a point to point communication link.

- Bit error rate is not the channel performance criteria, but rather packet loss statistics.
- Error detection is of limited interest. Integrity is assumed to be supported out-of-band by underlying detection at lower protocol layers.
- Explicit packet erasure side information is available
- Within the multicast group a single parity packet can correct more than one missing packet overall by correcting different single errors at different receivers

An erasure-based FEC code is applied as follows. Multicast transmission packets are formulated in a block of k packets at the source and transmitted across the network. The payloads of dropped packets (identified by missing sequence numbers at the receiver or by some other means) are treated as erasures in the FEC decoding process. It is desirable to have a code where the transmission of $k(data)+i(parity)$ frames of b bits in length allows any receiver successfully receiving any combination $N=k$ of these frames to decode the k frames of data. This property is true of Reed-Solomon (RS) codes with symbols over GF(2^b). RS codes are a class of non-binary FEC codes with excellent burst correction and erasure-filling properties [23]. In the following example design, we will assume the use of RS coding as our erasure code.

There are a number of engineering tradeoffs and variables beyond the scope of the overview given here which could be applied in the implementation of this method for multicast or unicast data transfer. For example, RS symbol sizes could be selected so that there was one RS symbol per packet. However, having large symbol sizes may result in an increase in decoder complexity. In the example design presented below, a RS code was chosen over Galois Field (2^8) resulting in codewords made of 8-bit symbols. This data symbol size is consistent with what current computers and networking protocols are designed to manipulate and transport. As a result, any maximum packet size (per transmission block) of L bytes in length can be used with L interleaved codewords per block. Erasure position information will be the same across all received codewords for linear block interleaving by this method. Figure 1 summarizes the method for formulating a given block of codewords for packet transmission at the encoder. As described above, receivers will use explicit information (e.g., sequence numbers) to detect lost packets and mark erasures for block decoding. And as noted, the data content size of individual packets may be less than L with zero-value padding bytes assumed for the untransmitted remainder of the packets (up to length L).

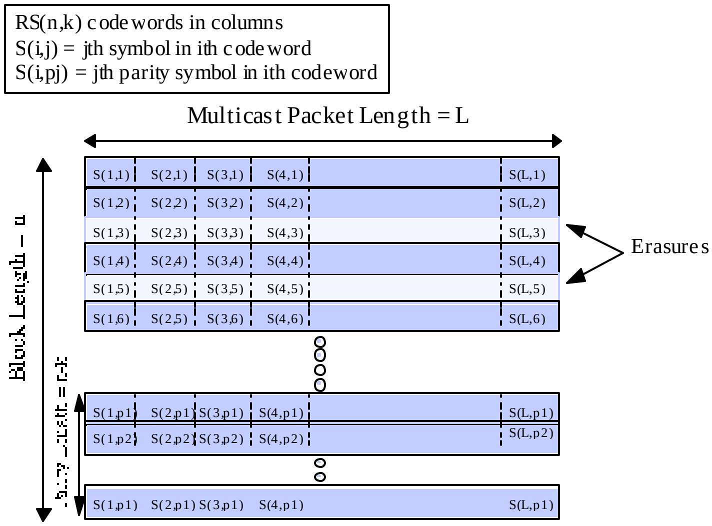

Figure 1: FEC block with Reed Solomon codes

As shown in Figure 1, Reed Solomon FEC codewords are constructed so that codeword symbols are striped across a number of successive *k* packets for a block of length *n*, where *n* is the total symbol length of the codeword. A RS encoder will be used to formulate *n-k* parity packets. The length of the block's largest packet (in bytes) will determine the number of FEC codewords interleaved within the block. Note that although the codewords are striped down across successive packets, transmission data ordering remains intact. Therefore, data can be immediately transmitted when ready and immediately used by an application as it is received, if needed. Data packets detected as missing at the receivers are marked as RS codeblock erasures.

At this point, there are a number of error recovery formulation approaches that can be considered. Additional parity information can be dynamically requested by the multicast receiver group after missing data is determined or alternatively a certain amount of parity information, either fixed or variable, can be provided along with the initial data packet transmission. We shall discuss the latter case first.

Once a receiver within the multicast group has determined the end of a data block transmission period, missing data packets can be detected through the use of sequence information provided along with the transmitted packets. At this point, the receiver has knowledge of the number of erasures within the received block and can request the appropriate number of parity packets to be transmitted. In order to provide the correct amount of parity to enable correction, the source need only know the largest number of missing packets among its receivers, *individually*. Receivers need only provide a single number (the number of missing packets in the block) to the source as part of a repair request. If the repair message suppression is desired than group receivers can use backoff timer windows to reduce the number and implosion effect of repair requests. Receivers need only respond if they have not heard a requested number of packets greater than or equal to their missing number for the corresponding block of transmitted data being requested. The benefit in bandwidth savings is twofold: the repair messages are simplified and the number of retransmitted packets is reduced since the same parity packets can repair *different* lost packets at *different* receivers as long as the total number of lost packets at the respective receivers is less than *n-k*. This is a powerful scalability concept for uncorrelated loss among large receiver groups. As an extreme example, if among a group of 100 multicast receivers 50 members lose 50 different packets within a RS multicast block the retransmission of a single parity packet to the group can correct all 50 missing packets. In addition, only a single repair request message is required: namely {reporting *x=1* packets missing for block *i*}. In this scenario, pure ARQ schemes would require 50 separate repair requests (e.g., missing sequence numbers) and 50 complete packet retransmissions since these are uncorrelated, different lost data packets. Within a multicast group, all packets are generally forwarded to all receivers with the multicast distribution tree and it follows that pure ARQ approaches produce wasteful retransmission of unwanted packets to all members when there is uncorrelated loss.

The case in which FEC parity is not transmitted with the original data block but is requested is called an integrated FEC design as opposed to a layered approach. Recall that with integrated FEC parity repair packets are interactively requested based upon the maximum number of lost packets at any one receiver amongst the receiver group. As repairing occurs in cycles following the original transmission block, it is instructive to examine the expected number of retransmission packets required within a repair cycles. In figure 2, these expected values are presented for the first repair cycle immediately following the initial transmission of the packet block. The results are based upon an uncorrelated probability of packet loss at each of the receivers and a block size of 20 packets was used. Expected retransmission requirements for receiver group sizes of 20, 100, and 1000 were calculated.

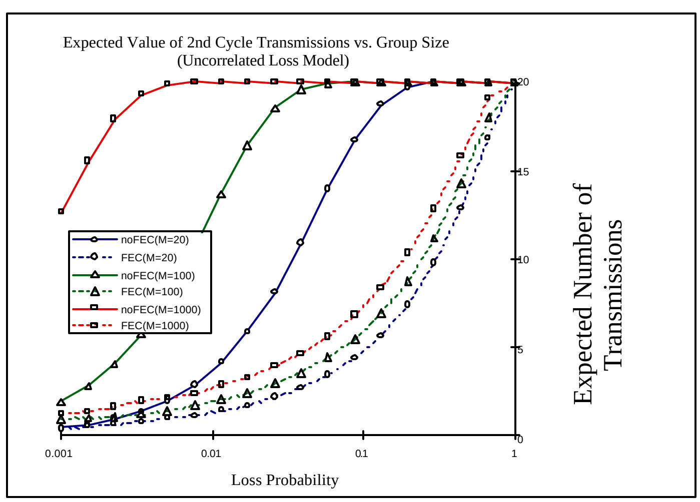

Figure 2: FEC vs. non-FEC Retransmission Requirements

Figure 2 demonstrates the significant increase in performance efficiency of FEC erasure-based ARQ approach. Even in the case of a relatively short block size (20) and a small group size (20), the number of retransmission repair packets is reduced from 4 to 2 for a 1% packet loss rate and from 17 to 5 for a 10% packet loss rate. As group size increases, figure 2 shows that FEC-based efficiency gain is significant even for small probabilities of

packet loss. Also, integrated FEC-based repair is never less bandwidth efficient than non-FEC ARQ. This is due to the fact that parity is transmitted only upon request and in the extreme case of completely correlated group loss achieves the same efficiency as non-FEC ARQ (i.e., the parity required for repair equals the number of packets lost).

Shortened RS Erasure Code Example

The following short example design helps illustrate how the integrated FEC/ARQ scheme can work in an actual reliable multicast protocol (e.g., MDP) where a number of processing tradeoffs are considered. This example also illustrates how short data transmission blocks can be supported.

For the purpose of the discussion, we will assume the use of IP multicast for bulk data transfer and can expect maximum packets in a rough range from 576 to 1500 bytes. We design our RS code around $b = 8$ bit symbols and use a RS(255,k) code as our basis. In order to decrease processing turnaround time for a block of received packets, we can shorten our code and reduce the transmission block size. Zero-filling x untransmitted data symbols at the encoder and the decoder forms a shortened RS codeword. This results in the transmission of only k-x codewords over the communication link. We choose k=235 and x=215 resulting in a shortened RS (40,20) code with 8-bit symbols. We will transmit data in blocks of 20 packets and allow the receivers to request up to 20 additional parity packets for erasure filling within these blocks. As shown in figure 2, for a group size of 100 and a receiver loss probability of 1% we require near 14 retransmitted packets for non-FEC and only around 2 parity packets for FEC for the first repair cycle. For uncorrelated packet loss, a non-FEC ARQ scheme is much less efficient in bandwidth usage and will also result in a longer delay to achieve repairing. For completely correlated packet loss, the transmission efficiency is equivalent since the maximum number of packets lost at a particular receiver equals the total lost amongst the receiver group. There is always an efficiency gain with this technique if the percentage of the packet loss among the receiver group is uncorrelated.

ISSUES

As mentioned previously, parity information may be supplied along with the original packet transmission rather than only requested in subsequent repair cycles. Or alternately, an independent layer below the reliable multicast protocol could provide FEC parity packets. This segregated approach has both advantages and disadvantages. The advantage is one of potentially further reducing initial repair request feedback messaging. The disadvantage is potential inefficient use of bandwidth if the coding rate and anticipated loss rate are not appropriately matched. The integrated approach does not suffer from this disadvantage since parity is interactively requested when needed. In the integrated case, the requirement for predicting channel loss rate statistics and/or adjusting the forward transmitted parity overhead of the protocol data blocks is minimized.

The codeword length in symbols affects the packet transmission block length and influences parity overhead ratio. Longer codewords (and thus longer FEC transmission blocks) result in increased delay in processing and the subsequent fill-in of any missing packets (erasures). However, for many applications an acceptable tradeoff may be attained between processing delay and parity packet transmission overhead. Another possibility is an adaptive approach where the amount of parity information provided per block is adjusted based upon measured multicast packet loss statistics. Further exploration of these design tradeoffs is considered future work.

A study in [6] presented performance modeling of both layered and integrated techniques described here under a number of FEC coding, group size, and error rate statistics. The conclusions demonstrate the significant potential performance improvements for integrated FEC/ARQ approaches to reliable multicasting over both layered FEC and non-FEC approaches.

Summary

We have discussed the issues of combining erasure-based FEC with reliable multicast transport ARQ schemes. It has been shown that an integrated FEC/ARQ for reliable multicast transport can achieve significant efficiency improvements, especially for uncorrelated packet loss. An example design of integrated this approach into an existing ARQ reliable multicast protocol was presented and performance tradeoffs were discussed.

Here is a summary of the features of an integrated FEC/ARQ approach applied to reliable multicasting:

- reduction in protocol bandwidth resources consumed (lower number of repair transmissions)
- reduction in the amount and size of repair request messages required
- improved scalability up to very large group sizes
- reasonable end system processing requirements

As number and size military group-based applications increases, reliable multicast dissemination will likely play an important role. The scalable and efficient features of an integrated multicast FEC/ARQ approach will improve performance dramatically over non-FEC multicast ARQ for error-prone wireless networks and congested military internetworks.

Acknowledgements

The author would like to thank R. Brian Adamson and Dr. Paul Crepeau for their discussions and contributions towards the ideas and models put forth in this paper.

References

[1] S. Floyd, V. Jacobson, S. McCanne, C. Liu, and L. Zhang. "A Reliable Multicast Framework for Light-weight Sessions and

Application Level Framing". In Proc. ACM SIGCOMM, August 1995.

[2] Macker, J.P., W.K. Dang, "The Reliable Dissemination Protocol", Internet Draft, October 1996, work in progress.

[3] Deng, Robert, "Hybrid ARQ Schemes for Point-to-Multipoint Communication over Nonstationary Broadcast Channels", IEEE Transactions on Communications, Vol. 41, No.9, September 1993.

[4] Rizzo, Luigi, "Effective Erasure Codes for Reliable Computer Communication Protocols", Technical Report, University of Pisa, Jan 9, 1997.

[5] Nonnenmacher, J., Ernst Biersack, "Reliable Multicast: Where to use FEC", Technical Report, Institut Eurocom.

[6] Nonnenmacher, J., Ernst Biersack, Towsley, D. "Parity-Based Loss Recovery for Reliable Multicast Transmission", Technical Report 97-17, Department of Computer Science, Univ of Mass, March 1997.

[7] Metzer, John, "An Improved Broadcast Retransmission Protocol", IEEE Transactions on Communications, Vol. Com-32, No.6, June 1984.

[8] Macker, J.P., M.S. Corson, "Controlled Link Sharing and Reliable Multicast for Asymmetric Networks", Pacific Telecommunications Conference Proceedings, Jan 1997.

[9] S. Deering. "Host Extensions for IP Multicasting". Internet RFC 1112, August 1989.

[10] Macker, J.P., M.S. Corson, E. Klinker, "Reliable Multicast Data Delivery for Military Internetworking", October 1996, IEEE MILCOM 96' Proceedings.

[11] Macker, J.P. " Controlled Link Sharing and Quality of Service for Military Internetworks", October 1996, IEEE Milcom 96' Proceedings.

[12] J. Chang and N. Maxemchuk. "Reliable Broadcast Protocols". ACM Transactions on Computer Systems, 2(3):251--275, August 1984.

[13] B. Whetton, T. Montgomery, and S. Kaplan. "A High Performance Totally Ordered Multicast Protocol". In Proc. INFOCOM, Boston, MA, March 1995.

[14] M. Kaashoek, A. Tannenbaum, Hummel, and Bal. "An Efficient Reliable Broadcast Protocol". Operating Systems Review, October 1989.

[15] S. Armstrong, A. Freier, and K. Marzullo. "Multicast Transport Protocol". Internet RFC 1301, February 1992.

[16] S. Pingali, D. Towsley, and J. Kurose. "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols". In Proc. INFOCOM, San Francisco, CA, October 1993.

[17] A. Koifman and S. Zabele. "RAMP: A Reliable Adaptive Multicast Protocol". In Proc. INFOCOM '96 (to appear), San Francisco, CA, 1996.

[18] R. Braudes and S. Sabele. "Requirements for Multicast Protocols". Internet RFC 1458, May 1993.

[19] B. Dempsey, M. Lucas, and A. Weaver. "Design and Implementation of a High Quality Video Distribution System using XTP Reliable Multicast". Technical report, Computer Science Department, University of Virginia, 1994.

[20] D. Clark and D. Tennenhouse. "Architectural Considerations for a New Generation of Protocols". In Proc. ACM SIGCOMM, pages 201--208, September 1990.

[21] H. Holbrook, S. Singhal, and D. Cheriton. "Log-Based Receiver-Reliable Multicast for Distributed Interactive Simulation". In Proc. ACM SIGCOMM, August 1995.

[22] R. Yavatkar, J. Griffioen, M. Sudan, "A Reliable Dissemination Protocol for Interactive Collaborative Applications," Technical Report, Department of Computer Science, University of Kentucky.

[23] Lin S, D.J. Costello, Error Control Coding, Prentice Hall, 1983.

[24] George Carle, "Towards Scalable Error Control for Reliable Multipoint Services in ATM Networks", In 12th International Conference on Computer Communications, ICCC'95, Seoul, Korea, August 1995.

[25] Matthias Grossglauser, "Optimal Deterministic Timeouts for Reliable Scalable Multicast", In Proceedings of INFOCOM '96, San Francisco, CA, March 1996.